REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* JANUARY 2013	**2. REPORT TYPE** JOURNAL ARTICLE (Pre Print)	**3. DATES COVERED** *(From - To)* JAN 2012 – APR 2014

4. TITLE AND SUBTITLE

A FAST AND ACCURATE ALGORITHM FOR ℓ1 MINIMIZATION PROBLEMS IN COMPRESSIVE SAMPLING

5a. CONTRACT NUMBER
IN-HOUSE

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
62788F

6. AUTHOR(S)

Feishe Chen (Syracuse University)
Lixin Shen (Syracuse University)
Yuesheng Xu (Syracuse University)
Bruce Suter (AFRL)

5d. PROJECT NUMBER
T2AW

5e. TASK NUMBER
IN

5f. WORK UNIT NUMBER
HO

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

PRIME
Air Force Research Laboratory/
Information Directorate
Rome Research Site/RITB
525 Brooks Road
Rome NY 13441-4505

SUB
Syracuse University
900 S Crouse Ave
Syracuse, NY 13210

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/Information Directorate
Rome Research Site/RITB
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TP-2013-037

12. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA Case Number: 88ABW-2013-0620
DATE CLEARED: 12 FEB 2013

13. SUPPLEMENTARY NOTES
One or more of the authors is a U.S. Government employee working within the scope of their Government job; therefore, the U.S. Government is joint owner of the work and has the right to copy, distribute, and use the work. All other rights are reserved by the copyright owner.GOOG

14. ABSTRACT

An accurate and efficient algorithm for solving the constrained ℓ1-norm minimization problem is highly needed and is crucial for the success of sparse signal recovery in compressive sampling. Most of existing algorithms in the literature give an approximate solution to the problem. We tackle the constrained ℓ1-norm minimization problem by reformulating it via an indicator function which describes the constraints. The resulting model is solved efficiently and accurately by using an elegant proximity operator based algorithm. We establish convergence analysis of the resulting algorithm. Numerical experiments show that the proposed algorithm performs well for sparse signals with magnitudes over a high dynamic range. Furthermore, it performs significantly better than the well-known algorithm NESTA in terms of the quality of restored signals and the computational complexity measured in the CPU-time consumed.

15. SUBJECT TERMS
Compressive Sampling, L1-norm, NESTA

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** MICHAEL LITTLE
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	13	**19b. TELEPHONE NUMBER (*Include area code)*** N/A

A Fast and Accurate Algorithm for ℓ_1 Minimization Problems in Compressive Sampling [*]

Feishe Chen[†] Lixin Shen[†] Bruce W. Suter[‡] Yuesheng Xu[†]

January 22, 2013

Abstract

An accurate and efficient algorithm for solving the constrained ℓ_1-norm minimization problem is highly needed and is crucial for the success of sparse signal recovery in compressive sampling. Most of existing algorithms in the literature give an approximate solution to the problem. We tackle the constrained ℓ_1-norm minimization problem by reformulating it via an indicator function which describes the constraints. The resulting model is solved efficiently and accurately by using an elegant proximity operator based algorithm. We establish convergence analysis of the resulting algorithm. Numerical experiments show that the proposed algorithm performs well for sparse signals with magnitudes over a high dynamic range. Furthermore, it performs significantly better than the well-known algorithm NESTA in terms of the quality of restored signals and the computational complexity measured in the CPU-time consumed.

1 Introduction

In this paper, we study recovery of an unknown vector $u_0 \in \mathbb{R}^n$ from the observed data $b \in \mathbb{R}^m$ and the model

$$b = Au_0 + z, \tag{1}$$

where A is a known $m \times n$ measurement matrix and $z \in \mathbb{R}^m$ is a noise term. Under an assumption that the vector u_0 of interest is sparse, the work in [4, 5] shows that an accurate estimation of u_0 is possible even when $m < n$, that is, the observations are less than unknowns. Recently, there is a significant body of work that focuses on finding an approximation to u_0 by solving a convex optimization problem. In the presence of noise-free data, i.e., $z = 0$, the optimization problem is

$$\text{(BP)} \qquad \min\{\|u\|_1 : u \in \mathbb{R}^n\} \quad \text{subject to} \quad b = Au,$$

which essentially is the basis pursuit problem proposed early in the context of time-frequency representation [8]. Here, $\|\cdot\|_1$ denotes the ℓ_1-norm of a vector in an Euclidean space. The optimization model (BP) can be solved by linear programming.

In the presence of noisy data, the linear constraint $b = Au$ in (BP) is relaxed to an inequality constraint $\|Au - b\|_2 \le \epsilon$, where $\|\cdot\|_2$ denotes the ℓ_2-norm of a vector in an Euclidean space. As a result, the optimization model (BP) becomes the basis pursuit denoising problem

$$(\text{BP}_\epsilon) \qquad \min\{\|u\|_1 : u \in \mathbb{R}^n\} \quad \text{subject to} \quad \|Au - b\|_2 \le \epsilon,$$

[*] This research is supported in part by 2012 Air Force Visiting Faculty Research Program, by an award from the NRC (National Research Council) via AFSOR, by the US National Science Foundation under grant DMS-1115523.

[†] Department of Mathematics, Syracuse University, Syracuse, NY 13244, USA.

[‡] Air Force Research Laboratory, AFRL/RITB, Rome, NY 13441-4505. Email: `Bruce.Suter@rl.af.mil`

where ϵ^2 is an estimated upper bound of the noise power.

Both problems (BP) and (BP$_\epsilon$) are closely related to the penalized least-squares problem

$$(\mathrm{QP}_\lambda) \qquad \min\left\{\frac{1}{2}\|Au-b\|_2^2+\lambda\|u\|_1 : u\in\mathbb{R}^n\right\}.$$

A large amount of research has been done on solving problems (BP), (BP$_\epsilon$), and (QP$_\lambda$). Here, we only give a brief and non-exhaustive review of results for these problems. In [8], problems (BP) and (QP$_\lambda$) are solved by first reformulating them as perturbed linear programming and then applying a primal-dual interior-point approach [24]. Recently, many iterative shrinkage/thresholding algorithms are proposed to handle problem (QP$_\lambda$). These include the proximal forward-backward splitting [9], the gradient projection for sparse reconstruction [12], the FISTA (fast iterative shrinkage-thresholding algorithm) [1], the fixed-point continuation algorithm [13], the Bregman iterative regularization [3, 25], and the reference therein. Problem (BP$_\epsilon$) also frequently appears in wavelet-based signal/image restoration [6, 7] with the matrix A associated with some inverse transforms.

Problem (BP$_\epsilon$) can be formulated as a second-order cone program and solved by interior-point algorithms. Many suggested algorithms for (BP$_\epsilon$) are based on repeatedly solving (QP$_\lambda$) for various values of λ. Such algorithms are referred to as the homotopy method originally proposed in [11, 20]. The homotopy method is also successfully applied to (BP) in [10]. A common approach for obtaining approximate solutions to (BP$_\epsilon$) is often accompanied by solving (QP$_\lambda$) for a decreasing sequence of values of λ [22]. The optimization theory asserts that problems (BP$_\epsilon$) and (QP$_\lambda$) are equivalent provided that the parameters ϵ and λ satisfy certain relationship [21]. Since this relationship is hard to compute in general, solving problem (BP$_\epsilon$) via repeatedly solving (QP$_\lambda$) for various values of λ is problematic. Recently, the NESTA [2] which employs Nesterov's optimal gradient method was proposed for solving relaxed versions of (BP) and (BP$_\epsilon$) via Nesterov's smoothing technique [18]. Clearly, the closeness of the solution to the relaxed version of (BP) (or the relaxed version of (BP$_\epsilon$)) to the solution to (BP) (or (BP$_\epsilon$)) is determined by the level of the closeness of the smoothed ℓ_1-norm to the ℓ_1-norm itself. Certainly, the performance of these approaches depends on the fine tuning of the parameter λ in (QP$_\lambda$) or a parameter that controls the degree of the closeness of the ℓ_1-norm and its smoothed version.

In this paper, we consider solving problems (BP) and (BP$_\epsilon$) by a different approach. We convert the constrained optimization problems to unified unconstrained one via an indicator function. The corresponding objective function for the unconstrained optimization problem is the sum of the ℓ_1-norm of the underlying signal u and the indicator function of a set in $\mathbb{R}^m$, which is $\{0\}$ for (BP) or the ϵ-ball for (BP$_\epsilon$), composing with the affine transformation $Au-b$. Non-differentiability of both the ℓ_1-norm and the indicator of the set imposes challenges for solving the associated optimization problem. Fortunately enough, their proximity operators have explicit expressions. The solutions for the problem can be viewed as fixed-points of a coupled equation formed in terms of these proximity operators. An iterative algorithm for finding the fixed-points is then developed. The main advantage of this approach is that solving (QP$_\lambda$) or smoothing the ℓ_1-norm are no longer necessary. This makes the proposed algorithm attractive for solving (BP) and (BP$_\epsilon$). The efficiency of fixed-point based proximity algorithms has been demonstrated in [9, 14, 15, 16] for various image processing models.

The rest of the paper is organized as follows: In section 2 we reformulate the ℓ_1-norm minimization problems (BP) and (BP$_\epsilon$) via an indicator function and characterize solutions of the proposed model in terms of fixed-point equations. We also point out the connection between the proposed model and (QP$_\lambda$) through the Moreau envelope. In section 3 we develop an algorithm for the resulting minimization problem based on the fixed-point equations arising from the characterization of the proposed model. Numerical experiments are presented in section 4. We draw our conclusions in section 5.

2 An ℓ_1-Norm Optimization Model via an Indicator Function

In this section, we consider a general optimization model that includes models (BP) and (BP$_\epsilon$) as its special cases and characterize solutions to the proposed model.

We begin with introducing our notation and recalling necessary background from convex analysis. For the usual d-dimensional Euclidean space denoted by $\mathbb{R}^d$ we define $\langle x, y\rangle := \sum_{i=1}^d x_i y_i$, for $x, y \in \mathbb{R}^d$, the standard inner product in $\mathbb{R}^d$. The class of all lower semicontinuous convex functions $f : \mathbb{R}^d \to (-\infty, +\infty]$ such that $\mathrm{dom} f := \{x \in \mathbb{R}^d : f(x) < +\infty\} \neq \emptyset$ is denoted by $\Gamma_0(\mathbb{R}^d)$. The indicator function of a closed convex set C in $\mathbb{R}^d$ is defined, at $u \in \mathbb{R}^d$, as

$$\iota_C(u) := \begin{cases} 0, & \text{if } u \in C, \\ +\infty, & \text{otherwise.} \end{cases}$$

Clearly, the indicator function ι_C is in $\Gamma_0(\mathbb{R}^d)$ for any closed nonempty convex set C. In particular, we define a ball in $\mathbb{R}^m$ centered at the origin with radius ϵ as $B_\epsilon := \{v : v \in \mathbb{R}^m \text{ and } \|v\|_2 \le \epsilon\}$.

Given a matrix $A \in \mathbb{R}^{m\times n}$ and a vector $b \in \mathbb{R}^m$, we consider the following optimization problem

$$\min\{\|u\|_1 + \iota_{B_\epsilon}(Au - b) : u \in \mathbb{R}^n\}. \tag{2}$$

We can easily see that if $\epsilon = 0$ then model (2) reduces to (BP) and if $\epsilon > 0$ then model (2) reduces to (BP$_\epsilon$). In other words, the both constrained optimization problems (BP) and (BP$_\epsilon$) can be unified as the unconstrained optimization problem (2) via the indicator function ι_{B_ϵ}.

In the following, we shall focus on characterizing solutions of model (2) using fixed-point equations. To characterize solutions of model (2), we first need two concepts, namely, the proximity operator and subdifferential of functions in $\Gamma_0(\mathbb{R}^d)$. For a function $f \in \Gamma_0(\mathbb{R}^d)$, the proximity operator of f with parameter λ, denoted by $\mathrm{prox}_{\lambda f}$, is a mapping from $\mathbb{R}^d$ to itself, defined for a given point $x \in \mathbb{R}^d$ by $\mathrm{prox}_{\lambda f}(x) := \mathrm{argmin}\left\{\frac{1}{2\lambda}\|u - x\|_2^2 + f(u) : u \in \mathbb{R}^d\right\}$. The subdifferential of a proper convex function $\psi \in \Gamma_0(\mathbb{R}^d)$ at a given vector $u \in \mathbb{R}^d$ is the set defined by

$$\partial\psi(u) := \{v : \ v \in \mathbb{R}^d \text{ and } \psi(w) \ge \psi(u) + \langle v, w - u\rangle \text{ for all } w \in \mathbb{R}^d\}.$$

The subdifferential and the proximity operator of the function ψ are related in the following way (see, e.g. [15]): For u in the domain of ψ and $v \in \mathbb{R}^d$

$$v \in \partial\psi(u) \text{ if and only if } u = \mathrm{prox}_\psi(u + v). \tag{3}$$

Now, with the help of the subdifferential and the proximity operator, we can characterize a solution of the indicator function based model (2) via fixed-point equations.

Proposition 2.1 *Let ϵ be a nonnegative number, let B_ϵ be the ball in $\mathbb{R}^m$ centered at the origin with radius ϵ, let b be a point in $\mathbb{R}^m$, and let A be an $m \times n$ matrix. If $u \in \mathbb{R}^n$ is a solution to model (2), then for any $\alpha > 0$ and $\beta > 0$ there exists a vector $v \in \mathbb{R}^m$ such that*

$$u = \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - \frac{\beta}{\alpha}A^\top v\right), \tag{4}$$

$$v = (I - \mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)})(Au + v). \tag{5}$$

Conversely, if there exist $\alpha > 0$, $u \in \mathbb{R}^n$, and $v \in \mathbb{R}^m$ satisfying equations (4) and (5), then u is a solution of model (2).

Proof: We first assume that $u \in \mathbb{R}^n$ is a solution of model (2). Set $\varphi := \iota_{B_\epsilon}(\cdot - b)$. Hence $Au - b$ must be in the ball B_ϵ. Therefore, both sets $\partial\|\cdot\|_1(u)$ and $\partial\varphi(Au)$ are nonempty. By Fermat's rule we have that $0 \in \partial\|\cdot\|_1(u) + A^\top\partial\varphi(Au)$. Therefore, for any $\alpha > 0$ and $\beta > 0$ there exist $w \in \frac{1}{\alpha}\partial\|\cdot\|_1(u)$ and $v \in \frac{1}{\beta}\partial\varphi(Au)$ such that $0 = \alpha w + \beta A^\top v$, i.e., $w = -\frac{\beta}{\alpha}A^\top v$. By using (3), inclusion $w \in \frac{1}{\alpha}\partial\|\cdot\|_1(u)$ implies $u = \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u + w)$, which is equation (4). Since $\frac{1}{\beta}\varphi = \varphi$ for any $\beta > 0$, inclusion $v \in \frac{1}{\beta}\partial\varphi(Au)$ leads to $Au = \mathrm{prox}_\varphi(v + Au)$, which is equivalent to (5).

Conversely, if equations (4) and (5) are satisfied for some $\alpha > 0$, $\beta > 0$, $u \in \mathbb{R}^n$, and $v \in \mathbb{R}^m$, using (3) again, we have that $-\frac{\beta}{\alpha}A^\top v \in \partial\left(\frac{1}{\alpha}\|\cdot\|_1\right)(u)$ and $v \in \partial\varphi(Au)$. Since $\partial\left(\frac{1}{\alpha}\|\cdot\|_1\right)(u) = \frac{1}{\alpha}\partial\|\cdot\|_1(u)$ and $\partial\varphi(Au) = \beta\partial\varphi(Au)$, we know from the above that $0 \in \partial\|\cdot\|_1(u) + A^\top\partial\varphi(Au)$. This indicates that u is a solution of model (2). The proof is complete. □

We remark that the above fixed-point characterization can be identified as a special case of Proposition 1 in [16]. We include the proof of Proposition 2.1 here for making the paper self-contained.

The proximity operators of the functions $\|\cdot\|_1$ and $\iota_{B_\epsilon}(\cdot - b)$ involved in the characterization can be computed efficiently. Indeed, the proximity operator $\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}$ is the soft-thresholding operator defined for $u \in \mathbb{R}^n$ by:

$$\left(\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u)\right)[i] = \max\left\{|u[i]| - \frac{1}{\alpha}, 0\right\}\mathrm{sign}(u[i]), \quad \text{for } i = 1, 2, \ldots, n. \tag{6}$$

The proximity operator $\mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)}$ is given by the following lemma.

Lemma 2.2 *Let ϵ be a nonnegative number, let B_ϵ be the ball in $\mathbb{R}^m$ centered at the origin with radius ϵ, let b be a point in $\mathbb{R}^m$. Then for a given $v \in \mathbb{R}^m$*

$$\mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)}(v) = b + \min\left\{1, \frac{\epsilon}{\|v - b\|_2}\right\} \cdot (v - b).$$

Proof: By the definition of the proximity operator, we can verify directly that $\mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)} = b + \mathrm{prox}_{\iota_{B_\epsilon}}(\cdot - b)$ and $\mathrm{prox}_{\iota_{B_\epsilon}}$ is the projection operator onto the ball B_ϵ. The result of this lemma follows immediately. □

We close this section by making a connection between the proposed model (2) and model (QP$_\lambda$) via the Moreau envelope introduced in [17]. Recall that the Moreau envelope of a function $f \in \Gamma_0(\mathbb{R}^d)$ with parameter μ at $x \in \mathbb{R}^d$ is $\mathrm{env}_{\mu f}(x) := \min\{\frac{1}{2\mu}\|y - x\|_2^2 + f(y) : y \in \mathbb{R}^d\}$, which is also in $\Gamma_0(\mathbb{R}^d)$. In particular, the Moreau envelope of the indicator function ι_{B_ϵ} with parameter μ is given in the following result.

Lemma 2.3 *Let ϵ be a nonnegative number, let μ be a positive number, let B_ϵ be the ball in $\mathbb{R}^m$ centered at the origin with radius ϵ. Then $\mathrm{env}_{\mu\iota_{B_\epsilon}}(x) = \frac{1}{2\mu}\left(\max\{\|x\|_2 - \epsilon, 0\}\right)^2$, for any $x \in \mathbb{R}^m$.*

Proof: By the definition of the Moreau envelope, we have that $\mathrm{env}_{\mu\iota_{B_\epsilon}}(x)$ is the square of the distance from the point x to the ball B_ϵ then scaled by the factor $\frac{1}{2\mu}$. Since this distance is either 0 if x is inside the ball or $\|x\|_2 - \epsilon$ if x is outside the ball, our result follows immediately. □

As an immediate consequence of Lemma 2.3 we have that $\mathrm{env}_{\lambda\iota_{\{0\}}} = \frac{1}{2\lambda}\|\cdot\|_2^2$. Hence, problem (QP$_\lambda$) can be rewritten in terms of the Moreau envelope as

$$\min\left\{\|u\|_1 + \mathrm{env}_{\lambda\iota_{\{0\}}}(Au - b) : u \in \mathbb{R}^n\right\}. \tag{7}$$

In recent literature, model (7) (i.e., (QP_λ)) is commonly adopted for finding an approximate solution to both problems (BP) and (BP_ϵ). Note that

$$\iota_{B_\epsilon}(x) - \mathrm{env}_{\lambda \iota_{B_\epsilon}}(x) = \iota_{B_\epsilon}(x)$$

and

$$\iota_{B_\epsilon}(x) - \mathrm{env}_{\lambda \iota_{\{0\}}}(x) = \begin{cases} -\frac{1}{2\lambda}\|x\|_2^2, & \text{if } x \in B_\epsilon; \\ +\infty, & \text{otherwise.} \end{cases}$$

These identities clearly indicate that $\mathrm{env}_{\lambda \iota_{B_\epsilon}}$ approximates ι_{B_ϵ} better than $\mathrm{env}_{\lambda \iota_{\{0\}}}$. Therefore, we comment that instead of model (7) the following model

$$\min\left\{\|u\|_1 + \mathrm{env}_{\lambda \iota_{B_\epsilon}}(Au - b) : u \in \mathbb{R}^n\right\} \tag{8}$$

would be more suitable for finding an approximate solution to problem (BP_ϵ). In other words, (QP_λ) used in many algorithms (e.g., [11, 20, 22, 23]) for (BP_ϵ) should be replaced by model (8). By doing so, an improvement for solving (BP_ϵ) should be expected if the noise power ϵ^2 can be estimated in advance. This will be studied in our future project. Unlike models (7) and (8) we solve model (2) directly without introducing the parameter λ.

3 An Algorithm and Its Convergence

In this section, we develop an algorithm for finding a solution of model (2) and provide a convergence analysis for the developed algorithm.

As we already know, all solutions of model (2) should satisfy the fixed-point equations given by (4) and (5). Our proposed algorithm for model (2) is derived based on the following equivalent form of equations (4) and (5)

$$\begin{cases} u = \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(\left(I - \frac{\beta}{\alpha}A^\top A\right)u - \frac{\beta}{\alpha}A^\top(v - w)\right), \\ w = \mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)}(Au + v), \\ v = Au + v - w. \end{cases} \tag{9}$$

Based on the above fixed-point equations in terms of u, w, and v, for arbitrary initial vectors $u^0 \in \mathbb{R}^n$, $w^0, v^0 \in \mathbb{R}^m$, we generate the sequence $\{u^k : k \in \mathbb{N}_0\}$ by the following iterative scheme

$$\begin{cases} u^{k+1} = \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(\left(I - \frac{\beta}{\alpha}A^\top A\right)u^k - \frac{\beta}{\alpha}A^\top(v^k - w^k)\right), \\ w^{k+1} = \mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)}(Au^{k+1} + v^k), \\ v^{k+1} = Au^{k+1} + v^k - w^{k+1}, \end{cases} \tag{10}$$

where $\mathbb{N}_0 := \{0, 1, \ldots\}$.

To show convergence of the iterative scheme (10), we recall a result from [14].

Lemma 3.1 (Theorem 3.5 in [14]) *If x is a vector in $\mathbb{R}^n$, A is an $m \times n$ matrix, φ is in $\Gamma_0(\mathbb{R}^m)$ and α, β, λ are positive numbers such that $\frac{\beta}{\lambda\alpha} < \frac{1}{\|A\|^2}$, then the sequence $\{u^k : k \in \mathbb{N}_0\}$ generated by the following iterative scheme*

$$\begin{cases} u^{k+1} = x + \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(\left(I - \frac{\beta}{\lambda\alpha}A^\top A\right)u^k - x - \frac{\beta}{\lambda\alpha}A^\top(v^k - w^k)\right), \\ w^{k+1} = \mathrm{prox}_{\frac{1}{\beta}\varphi}(Au^{k+1} + v^k), \\ v^{k+1} = Au^{k+1} + v^k - w^{k+1} \end{cases} \tag{11}$$

converges to a solution of the optimization problem

$$\min\{\lambda\|u - x\|_1 + \varphi \circ A(u) : u \in \mathbb{R}^m\}. \tag{12}$$

With the help of Lemma 3.1, the following result shows that under appropriate conditions on parameters α and β the sequence $\{u^k : k \in \mathbb{N}_0\}$ converges to a solution of model (2).

Theorem 3.2 *Let ϵ be a nonnegative number, let B_ϵ be the ball in $\mathbb{R}^m$ centered at the origin with radius ϵ, let b be a point in $\mathbb{R}^m$, and let A be an $m \times n$ matrix. If*

$$\frac{\beta}{\alpha} < \frac{1}{\|A\|^2}, \tag{13}$$

then for arbitrary initial vectors $u^0 \in \mathbb{R}^n$, $w^0, v^0 \in \mathbb{R}^m$, the sequence $\{u^k : k \in \mathbb{N}_0\}$ generated by the iterative scheme (10) *converges to a solution of model* (2).

Proof: By setting $x = 0$ and $\lambda = 1$ and identifying $\varphi = \iota_{B_\epsilon}(\cdot - b)$ in model (12), the iterative scheme (10) can be viewed as a special case of the one given in (11). The desired result follows immediately from Lemma 3.1. □

The convergence result given by Theorem 3.2 offers a practical way to find a solution of model (2). Since the explicit forms of the proximity operators $\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}$ and $\mathrm{prox}_{\iota_{B_\epsilon}(\cdot - b)}$ are given by (6) and Lemma 2.2, respectively, based on Theorem 3.2 a unified approach for solving both (BP) and (BP$_\epsilon$) is depicted in Algorithm 1.

Algorithm 1 (The iterative scheme for model (BP$_\epsilon$) with $\epsilon \geq 0$)

```
Initialization: $v^0 \in \mathbb{R}^m$, $w^0 \in \mathbb{R}^m$, $u^0 \in \mathbb{R}^n$, $\epsilon > 0$, $\alpha > 0$, and $\beta > 0$ with $\frac{\beta}{\alpha} < \frac{1}{\|A\|^2}$.
repeat($k \geq 0$)
    Step 1: $u^{k+1} \leftarrow \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(\left(I - \frac{\beta}{\alpha}A^\top A\right)u^k - \frac{\beta}{\alpha}A^\top(v^k - w^k)\right)$
    Step 2: $w^{k+1} \leftarrow \begin{cases} Au^{k+1} + v^k, & \text{if } \|Au^{k+1} + v^k - b\|_2 < \epsilon; \\ b + \epsilon\frac{Au^{k+1} + v^k - b}{\|Au^{k+1} + v^k - b\|_2}, & \text{otherwise.} \end{cases}$
    Step 3: $v^{k+1} \leftarrow Au^{k+1} + v^k - w^{k+1}$
until a given stopping criteria is met
```

We make some comments on Algorithm 1. Step 1 of computing u^{k+1} is from the first equation in (10); Step 2 of computing w^{k+1} is from the second equation in (10) and Lemma 2.2; Step 3 of computing v^{k+1} is exactly the same as the last equation in (10). This algorithm can be presented in a more computationally efficient way by combining Step 2 and Step 3 together and eliminating the intermediate variable w^k. The motivation comes from the observation $Au^k - w^k = v^k - v^{k-1}$ which is due to the third step of Algorithm 1. Substituting $Au^k - w^k$ in Step 1 by $v^k - v^{k-1}$ yields that

$$u^{k+1} = \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k - \frac{\beta}{\alpha}A^\top(2v^k - v^{k-1})\right), \tag{14}$$

with an assumption $v^{-1} = v^0 - (Au^0 - w^0)$ for given u^0, w^0, and v^0. We can further substitute w^{k+1} computed in Step 2 into Step 3. In this way, the intermediate variable w^k is no longer needed. Hence, these simplifications yield Algorithm 2, a variant of Algorithm 1. When $\epsilon = 0$, all vectors w^{k+1} in Algorithm 1 are equal to the constant vector b for all $k \geq 0$. Because of this, we would like to set $w^0 = b$ in both Algorithms 1 and 2. Finally, it is more efficient to update

u^{k+1} with Step 1 of Algorithm 2 than Step 1 of Algorithm 1 in each iteration since the matrix-vector multiplication involving A is not required in equation (14). However, updating u^{k+1} via the formulation of Step 2 in Algorithm 1 can be implemented through the use of the component-wise Gauss-Seidel iteration which may accelerate the rate of convergence of the algorithm and therefore reduce the total CPU-time consumed. The efficiency of component-wise Gauss-Seidel iteration has been verified in [14, 15].

Algorithm 2 (A variant of Algorithm 1 for model (BP$_\epsilon$))

```
Initialization: $v^0 \in \mathbb{R}^m$, $u^0 \in \mathbb{R}^n$, $\epsilon > 0$, $\alpha > 0$, and $\beta > 0$ with $\frac{\beta}{\alpha} < \frac{1}{\|A\|^2}$; set $v^{-1} = v^0 - (Au^0 - d^0)$,
repeat($k \geq 0$)
    Step 1: $u^{k+1} \leftarrow \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k - \frac{\beta}{\alpha}A^\top(2v^k - v^{k-1})\right)$
    Step 2: $v^{k+1} \leftarrow \begin{cases} 0, & \text{if } \|Au^{k+1} + v^k - b\|_2 < \epsilon; \\ \left(1 - \frac{\epsilon}{\|Au^{k+1}+v^k-b\|_2}\right)(Au^{k+1} + v^k - b), & \text{otherwise.} \end{cases}$
until a given stopping criteria is met
```

4 Numerical Simulations

This section is devoted to showing the numerical performance of the proposed algorithms for compressive sampling. We use the algorithm NESTA [2] as a comparison. We focus on sparse signals with various dynamic ranges and measurement matrices from randomly partial discrete cosine transforms (DCTs) and evaluate performance of algorithms in terms of various error metrics, speed, and robustness to noise. All the experiments are performed in Matlab 7.11 on Thinkpad T400 with Intel Core Duo CPU @2.26G, 3GB RAM on Windows Vista Home Basic operating system.

We begin with a description of generating the $m \times n$ sensing matrix A and length-n and s-sparse signals. In each experimental trial, the $m \times n$ sensing matrix A is generated by randomly picking m rows from the $n \times n$ DCT matrix. Sparse signals u used in our experiments are generated according to [2]. In each experimental trial, a length-n, s-sparse signal (a signal having exactly s nonzero components), is generated in such a way that non-zero components are given by

$$\eta_1 10^{\theta\eta_2}, \tag{15}$$

where $\eta_1 = \pm 1$ with probability 1/2 and η_2 is uniformly distributed in $[0, 1]$. The locations of the nonzero components are randomly permuted. Clearly, the range of the magnitude of nonzero components of an s-sparse signal is $[0, 10^\theta]$ with the parameter θ controlling this dynamic range. An observed signal (data) is collected by $b = Au + z$, where z represents a Gaussian noise.

The accuracy of a solution obtained from a specific algorithm is quantified by the relative ℓ_2-error, the relative ℓ_1-error, and the absolute ℓ_∞-error defined, respectively, as follows:

$$\|u - u_\diamond\|/\|u\|, \quad |\|u\|_1 - \|u_\diamond\|_1|/\|u\|_1, \quad \|u - u_\diamond\|_\infty, \tag{16}$$

where u is the true data and $u_\diamond$ is the restored data. All results reported in the tables of this section are the means and the standard deviations of these relative errors from simulations that were performed 20 trials.

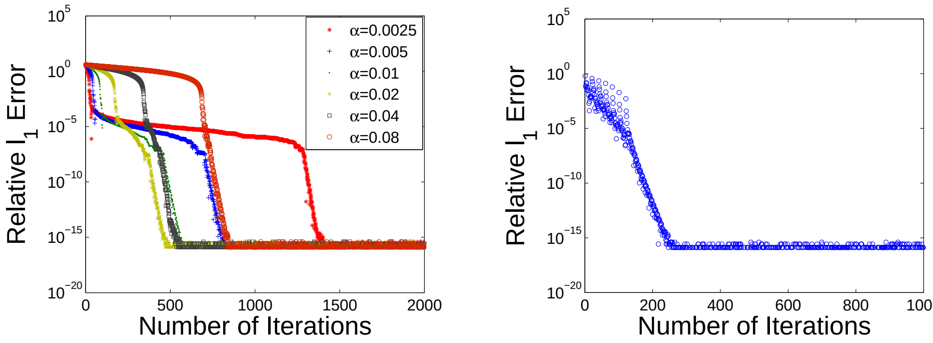
To use Algorithm 2, one needs to fix the parameters α and β such that $\beta/\alpha < 1/\|A\|^2$ (see Theorem 3.2). From Step 1 of the algorithm, the ratio β/α plays a role of step-size of changing u^k.

Therefore, we keep this ratio as big as possible and set

$$\beta = \frac{0.999}{\|A\|^2}\alpha \tag{17}$$

in our numerical experiments. In such the way, α is essentially the only parameter that needs to be determined. We now investigate the impact of the parameter α on the performance of Algorithm 2.

To investigate the impact of varying the parameter α on the performance of Algorithm 2, we consider the configuration of $n = 2^{15}$, $m = n/2$, $s = 0.05$ and the dynamic range parameter $\theta = 5$. The observed data is noise free. Six different values of α, namely, 0.0025, 0.005, 0.01, 0.02, 0.04, and 0.08, are tested. Figure 4.1(a) depicts the traces of the relative ℓ_1-error (see (16)) against the number of iterations for each α. As it can be seen from this figure, for $\alpha = 0.0025$, the smallest value in our test, the relative ℓ_1-error drops rapidly from 1 to 10^{-4}, stabilizes with insignificant changes for about 1200 iterations and then quickly drops again to the level of 10^{-16}. When α increases from 0.0025 to 0.08, the number of iterations required for the relative ℓ_1-error dropping from 1 to 10^{-4} increases. Meanwhile, the numbers of iterations for the transitions from the first sharp jump region to the second one decrease. For example, it is about 700 for $\alpha = 0.005$ and only few iterations for $\alpha = 0.08$. These observations motivate us to extend Algorithm 2 to a scenario in which the parameter α can be updated during the iteration with the goal of reducing the number of iterations. The proposed approach is rather simple. It begins with a relative small α and then increases it for every given amount of iterations. A detailed flow of this new approach is given in Algorithm 3.



(a) (b)

Figure 4.1: The relative ℓ_1 error (the vertical axis with a base 10 logarithmic scale) versus number of iterations (the horizontal axis). (a) Convergence of Algorithm 2 with different values of α; (b) Convergence of Algorithm 3.

Three new parameters introduced in Algorithm 3 are integers $p > 0$, $\tau > 1$, and $T > 0$. The parameter T is the allowable maximum number of updating the parameters α and β. For each update, the pair (α, β) will change to $(\tau\alpha, \beta/\tau)$ that will keep the ratio β/α unchanged. The parameter p is to indicate that the underlying algorithm with a pair (α, β) will iterate p times before the algorithm with the pair $(\tau\alpha, \beta/\tau)$ runs another p times. We now demonstrate the efficiency of varying the parameters α and β via applying Algorithm 3 for the same data used in Figure 4.1(a). We set $T = 6$, $\tau = 4$, and $p = 20$ and initialize $\alpha = \frac{m}{n}\frac{20}{\|A^\top b\|_\infty}$. Again, we choose β by using (17). The corresponding result is shown in Figure 4.1(b). It is clear to see that it takes about 200 iterations to drop the relative ℓ_1 error down below 10^{-14}. Hence the strategy of updating the parameters α and β as described in Algorithm 3 is reasonable.

Algorithm 3 (A variant of Algorithm 1 for model (BP_ϵ))

```
Given: integers $p > 0$, $\tau > 1$, and $T > 0$; $\epsilon > 0$
Initialization: $v^0 \in \mathbb{R}^m$, $u^0 \in \mathbb{R}^n$, $\alpha > 0$, and $\beta > 0$ with $\frac{\beta}{\alpha} < \frac{1}{\|A\|^2}$; set $v^{-1} = v^0 - (Au^0 - d^0)$
repeat($k \geq 0$)
    Step 1: Compute $u^{k+1}$ using Step 1 of Algorithm 2
    Step 2: Compute $v^{k+1}$ using Step 2 of Algorithm 2
    Step 3: If $k$ is a multiple of $p$ and the number of changing the parameters $\alpha$ and $\beta$ does not exceed $T$, update $\alpha \leftarrow \tau\alpha$, $\beta \leftarrow \frac{\beta}{\tau}$
until a given stopping criteria is met
```

The rest of this section consists of two parts. Part one contains the results for (BP) while part two contains the results for (BP_ϵ).

In part one, we compare the performance of Algorithm 3 with that of the NESTA [2]. The algorithm NESTA was developed by applying a smoothing technique for the nonsmooth ℓ_1-norm and an accelerated first-order scheme, both from Nesterov's work [19]. A parameter denoted by μ is used to control how close the smoothed ℓ_1-norm to the ℓ_1-norm will be. Two different levels of μ, namely, $\mu = 10^{-2}$ and $\mu = 10^{-7}$, are used for the NESTA in experiments. A parameter δ for tolerance in the NESTA varies for different values of the smoothing parameter μ and needs to be determined. We finally choose $\delta = 10^{-8}$ for $\mu = 10^{-2}$ and $\delta = 10^{-12}$ for $\mu = 10^{-7}$ since such choices lead to reasonable results. For Algorithm 3, we set $p = 20$ and T to be the smallest integer that is greater than $\log_{10}(\frac{n}{m}\|A^\top b\|_\infty)$. In our experiments, we notice that $\frac{n}{m}\|A^\top b\|_\infty$ is approximately equal to 10^θ. As a result, T is about $\theta + 1$. The stopping criterion of Algorithm 3 is that the relative errors between the successive iterates of the reconstructed signal should satisfy the inequality $\|u^{k+1} - u^k\|/\|u^k\| < 10^{-15}$.

In our experiments for problem (BP), the dimensions n and the dynamic ranges θ of the unknown signals are chosen from $\{2^{13}, 2^{15}, 2^{17}\}$ and $\{1, 3, 5\}$, respectively. The number of nonzero entries s is set to be $0.05n$, $0.02n$, $0.01n$ respectively for the number of measurements $m = n/2$, $n/4$, $n/8$. The left-column of Figure 4.2 shows the results of Algorithm 3 and the NESTA when the dimension of the tested signals n is 2^{17} and the number of measurements m is $n/4$. The symbols '□', '♢', and '∇' denote the results produced by Algorithm 3, the NESTA with $\mu = 10^{-2}$, and the NESTA with $\mu = 10^{-7}$, respectively. The colors 'red', 'blue', and 'yellow' represent the dynamic ranges of the tested signals with θ being 1, 3, and 5, respectively. Three error metrics, namely the relative ℓ_2-error, ℓ_1-error, ℓ_∞-error, are respectively displayed with a base 10 logarithmic scale plot for the vertical axis. We see clearly that Algorithm 3 is significantly better than the NESTA in terms of these error metrics for the unknown signals with various dynamic ranges and the CPU time consumed. We also observe that the relative ℓ_2-error and ℓ_1-error of the results recovered by Algorithm 3 along with the CPU time consumed are quite robust with respect to the dynamic ranges of the unknown signals. The same conclusions can be drown for the results with $m = n/2$ and $m = n/8$ as well.

Part two presents results for problem (BP_ϵ). The settings of dimension, sparsity, and dynamic range of unknown signals for problem (BP_ϵ) are the same as those for problem (BP). The only difference is that measurements in part two are contaminated with noise. In our experiments, noise levels in the measurements vary with the dynamic ranges of the unknown signals. More precisely, the noise levels σ are set to be 0.05, 1.0, and 5.0 corresponding to choices 1, 3, and 5 of the dynamic range parameter θ, respectively. It turns out that the noise power is $\epsilon^2 = m\sigma^2$. The stopping criteria for our algorithm is that the relative errors between the successive iterates of the reconstructed signal should satisfy the inequality $\|u^{k+1} - u^k\|/\|u^k\| < 10^{-5}$. For the smoothing

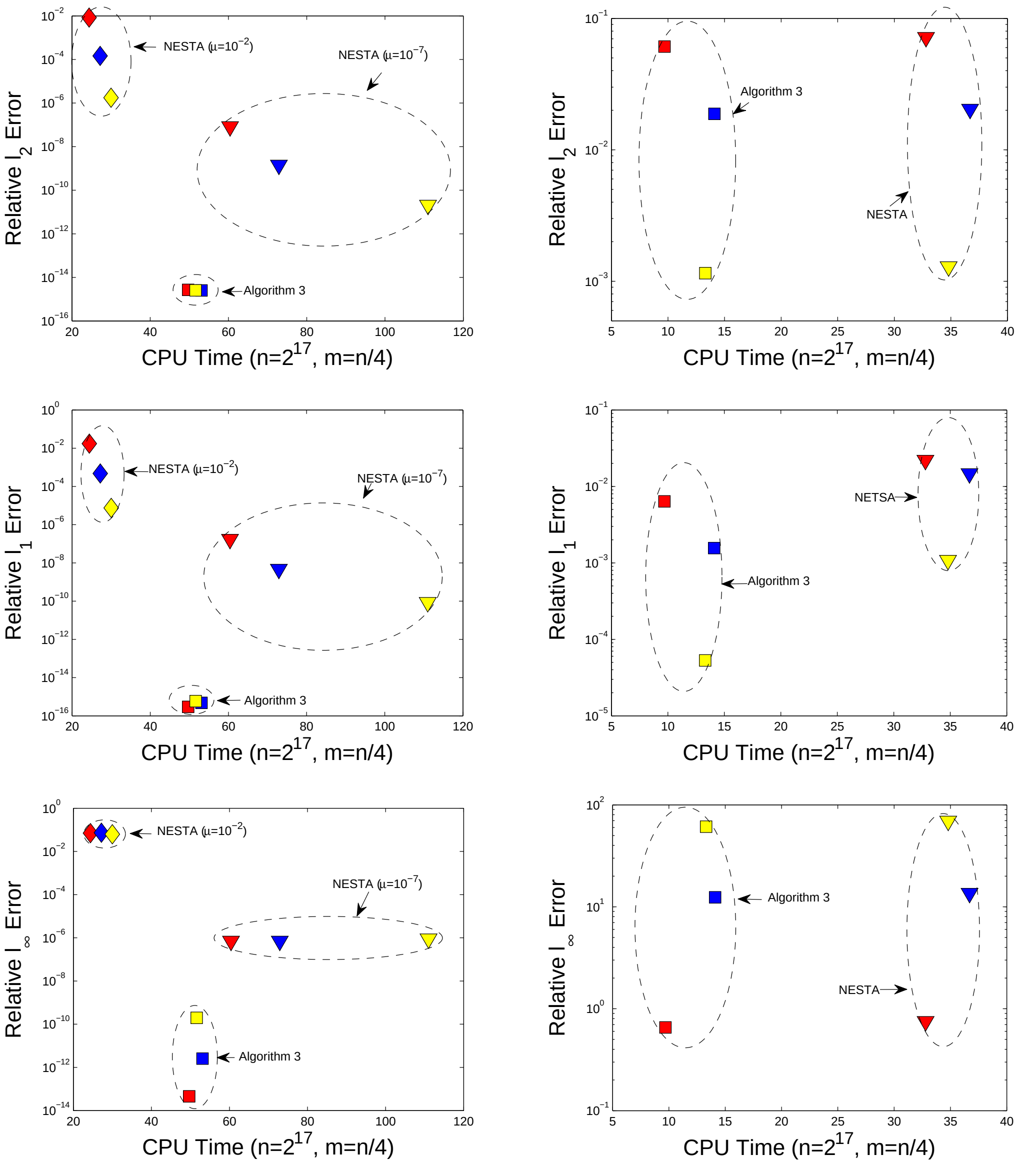


Figure 4.2: The relative ℓ_1, ℓ_2, and ℓ_∞ errors (the vertical axis with base 10 logarithmic scale) versus the CPU time consumed (the horizontal axis) for (the left-column) the noise free case and (the right-column) the noise case. The colors 'red', 'blue', and 'yellow' represent the dynamic ranges of the tested signals with θ being 1, 3, and 5, respectively.

parameter μ in the NESTA, we choose the default setting $\mu = \max\{0.1\sigma, 0.01\}$. The right column of Figure 4.2 shows the results of Algorithm 3 and the NESTA when the dimension of the tested signals n is 2^{17} and the number of measurements m is $n/4$. We see that the accuracy of our algorithm is better than that of the NESTA in terms of relative ℓ_2-error and relative ℓ_∞-error. There is up to 1 order of improvement of accuracy regarding the relative ℓ_1-error with Algorithm 3. Regarding the consumed CPU time, Algorithm 3 consumes at most half of the CPU time that the NESTA does. Again, the same conclusions can be drown for the results with $m = n/2$ and $m = n/8$ as well.

5 Conclusions

We reformulated the ℓ_1-norm minimization problems (BP) and (BP$_\epsilon$) via indicator functions as unconstrained minimization problems. The objective function for each unconstrained problem is the sum of the ℓ_1-norm of the underlying signal u and the indicator function of a set in $\mathbb{R}^m$, which is $\{0\}$ for (BP) or the ϵ-ball for (BP$_\epsilon$), composing with the affine transformation $Au - b$. Due to the structure of this objective function and the availability of the explicit forms of the proximity operators for both the ℓ_1-norm and the indicator function, an accurate and efficient algorithm is developed for recovering sparse signals based on fixed-point equation. The algorithm outperforms the state-of-the-art algorithm NESTA in terms of the relative ℓ_2, the relative ℓ_1, and the absolute ℓ_∞ error measures as well as the CPU time for tested signals ranging from a low dynamic range to a high dynamic range with different sizes.

References

[1] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 183–202.

[2] S. Becker, J. Bobin, and E. Candes, *NESTA: a fast and accurate first-order method for sparse recovery*, SIAM Journal on Imaging Sciences, 4 (2009), pp. 1–39.

[3] J.-F. Cai, S. Osher, and Z. Shen, *Split Bregman methods and frame based image restoration*, Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal, 2 (2009), pp. 337–369.

[4] E. Candes, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Communications on Pure and Applied Mathematics, 59 (2006), pp. 1207–1223.

[5] E. Candes and T. Tao, *Near optimal signal recovery from random projections: Universal encoding strategies?*, IEEE Transactions on Information Theory, 52 (2006), pp. 5406–5425.

[6] A. Chambolle, R. DeVore, N.-Y. Lee, and B. Lucier, *Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage*, IEEE Transactions on Image Processing, 7 (1998), pp. 319–335.

[7] R. Chan, T. Chan, L. Shen, and Z. Shen, *Wavelet algorithms for high-resolution image reconstruction*, SIAM Journal on Scientific Computing, 24 (2003), pp. 1408–1432.

[8] S. Chen, D. Donoho, and M. Saunders, *Atomic decomposition by basis pursuit*, SIAM Journal of Scientific Computing, 20 (1998), pp. 33–61.

[9] P. Combettes and V. Wajs, *Signal recovery by proximal forward-backward splitting*, Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal, 4 (2005), pp. 1168–1200.

[10] D. Donoho and Y. Tsaig, *Fast solution of ℓ_1-norm minimization problems when the solution may be sparse*, IEEE Transactions on Information Theory, 54 (2008), pp. 4789–4812.

[11] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, *Least angle regression*, The Annals of Statistics, 32 (2004), pp. 407–451.

[12] M. Figueiredo, S. Wright, and R. Nowak, *Gradient projection for sparse reconstruction: Applications to compressed sensing and other inverse problems*, IEEE Journal of Selected Topics in Signal Processing, 1 (2007), pp. 586–597.

[13] E. Hale, W. Yin, and Y. Zhang, *Fixed-point continuation for ℓ_1 minimization: Methodology and convergence*, SIAM Journal on Optimization, 19 (2008), pp. 1107–1130.

[14] Q. Li, C. A. Micchelli, L. Shen, and Y. Xu, *A proximity algorithm accelerated by Gauss-Seidel iterations for* L1/TV *denoising models*, Inverse Problems, 28 (2012), p. 095003.

[15] C. A. Micchelli, L. Shen, and Y. Xu, *Proximity algorithms for image models: Denoising*, Inverse Problems, 27 (2011), p. 045009(30pp).

[16] C. A. Micchelli, L. Shen, Y. Xu, and X. Zeng, *Proximity algorithms for image models II: L1/TV denosing*, Advances in Computational Mathematics, online version available, (2011).

[17] J.-J. Moreau, *Fonctions convexes duales et points proximaux dans un espace hilbertien*, C.R. Acad. Sci. Paris Sér. A Math., 255 (1962), pp. 1897–2899.

[18] Y. Nesterov, *Smooth minimization of non-smooth functions*, Mathematical Programming, 103 (2005), pp. 127–152.

[19] ——, *Smooth minimization of non-smooth functions*, Mathematical Programming, Series A, 103 (2005), pp. 127–152.

[20] M. Osborne, B. Presnell, and B. Turlach, *A new approach to variable selection in least squares probloems*, IMA Journal on Numerical Analysis, 20 (2000), pp. 389–403.

[21] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[22] E. van den Berg and M. P. Friedlander, *Probing the pareto frontier for basis pursuit solutions*, SIAM Journal on Scientific Computing, 31 (2008), pp. 890–912.

[23] ——, *Sparse optimization with least-squares constraints*, SIAM Journal on Optimization, 21 (2011), pp. 1201–1229.

[24] S. J. Wright, *Primal-Dual Interior-Point Methods*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[25] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, *Bregman iterative algorithms for ℓ^1 minimization with applications to compressed sensing*, SIAM Journal on Imaging Sciences, 1 (2008), pp. 143–168.